**Using assign statement**

module fourto1 (a,select,out);

input [3:0]a;

input [1:0]select;

output out;

assign out=a[select];

endmodule

module sixteento1 (A,S,Out);

input [15:0] A;

input [3:0] S;

output Out;

wire [3:0]O;

fourto1 F1 (A[3:0],S[1:0],O[0]);

fourto1 F2 (A[7:4],S[1:0],O[1]);

fourto1 F3 (A[11:8],S[1:0],O[2]);

fourto1 F4 (A[15:12],S[1:0],O[3]);

fourto1 F5 (O[3:0],S[3:2],Out);

endmodule

**Using case statement**

module fourto1 (a,select,out);

input [3:0]a;

input [1:0]select;

output reg out;

always@(\*)

begin

case(select)

2'b00: out = a[0];

2'b01: out = a[1];

2'b10: out = a[2];

2'b11: out = a[3];

default: out = 1'bx;

endcase

end

endmodule

module sixteento1 (A,S,Out);

input [15:0] A;

input [3:0] S;

output Out;

wire [3:0]O;

fourto1 F1 (A[3:0],S[1:0],O[0]);

fourto1 F2 (A[7:4],S[1:0],O[1]);

fourto1 F3 (A[11:8],S[1:0],O[2]);

fourto1 F4 (A[15:12],S[1:0],O[3]);

fourto1 F5 (O[3:0],S[3:2],Out);

endmodule

**Using if else statement**

module fourto1 (a,select,out);

input [3:0]a;

input [1:0]select;

output reg out;

always@(\*)

if(select == 2'b00)

out = a[0];

else if (select == 2'b01)

out = a[1];

else if (select == 2'b10)

out = a[2];

else if (select == 2'b11)

out = a[3];

endmodule

module sixteento1 (A,S,Out);

input [15:0] A;

input [3:0] S;

output Out;

wire [3:0]O;

fourto1 F1 (A[3:0],S[1:0],O[0]);

fourto1 F2 (A[7:4],S[1:0],O[1]);

fourto1 F3 (A[11:8],S[1:0],O[2]);

fourto1 F4 (A[15:12],S[1:0],O[3]);

fourto1 F5 (O[3:0],S[3:2],Out);

endmodule

**Testbench common to all the implementation**

module testbench;

reg [15:0]A;

reg [3:0]S;

wire Result;

sixteento1 S1 (A,S,Result);

initial

begin

$dumpfile("waveform.vcd");

$dumpvars(0,testbench);

$monitor($time,"A=%h,S=%h,Result=%b",A,S,Result);

#5 A=16'ha067; S=4'h0;

#5 S=4'h1;

#5 S=4'h2;

#5 S=4'h3;

#5 S=4'h4;

#5 S=4'h5;

#5 S=4'h6;

#5 S=4'h7;

#5 S=4'h8;

#5 S=4'h9;

#5 S=4'ha;

#5 $finish;

end

endmodule